Self-Supervised Terrain Classification for Planetary Rovers

Christopher A. Brooks, Karl Iagnemma
Department of Mechanical Engineering
Massachusetts Institute of Technology
77 Massachusetts Ave.
Cambridge, MA 02139
{cabrooks, kdi}@mit.edu

***Abstract*—For planetary rovers, autonomous mobility is a key to enabling greater science return. While current terrain sensing approaches can be used to detect geometric hazards, such as rocks and cliffs, they are limited in their ability to detect non-geometric hazards, such as loose sand in which a rover may become entrenched. This paper presents a self-supervised classification approach to learning the visual appearance of terrain classes which relies on vibration-based sensing of wheel-terrain interaction to identify these terrain classes. Experiments with a four-wheeled rover in Mars-analog terrain demonstrate the potential for this approach.**

I. Introduction and Related Work

The ability of mobile robots to explore other planets is increasingly dependent on the autonomous mobility capabilities of these robots. Targets of scientific interest for future missions include terrains such as craters, ravines, and cliffs, where landing is treacherous. Therefore, a planetary rover must land at a distant site then travel to its target. Limits on communication with observers on Earth mean that close supervision can put significant restrictions on the distance a rover can travel during a mission lifetime. Thus, advances in robot autonomy will lead to payoffs in terms of scientific return.

One current limitation in autonomous mobility lies in terrain sensing. Given an accurate map of the ease of traversability of terrain, existing path planning algorithms can generate a route to a target which avoids known obstacles [1,2,3]. The ability to generate this map, by remotely detecting possible hazards, would enable safe traversal of previously unexplored rough terrain.

While geometric hazards, such as large rocks, can be sensed remotely using stereo vision [4], little research has addressed remote sensing of non-geometric hazards, such as the loose drift material in which the Mars Exploration Rover (MER) Opportunity became entrenched in April 2005. Non-geometric hazards are highly dependent on wheel-terrain interaction, so the sensing of such hazards has focused on using the rover wheels as sensors. Examples include wheel sinkage measurement [5,6], parametric soil characterization [7], wheel slip detection [8] and explicit traversability estimation [9]. This sensing is inherently local—the rover wheel must be in contact with the terrain to make a measurement—so it is of limited use for hazard avoidance.

Remotely sensing non-geometric hazards depends on extrapolating from locally sensed data to gain information about more distant terrain, and this has only recently received attention from researchers. One group has attempted to correlate terrain appearance with wheel slip using a combination of unsupervised clustering and model fitting [10]. Another group has attempted to distinguish traversable terrain from non-traversable terrain using a self-supervised framework, to reduce the false positive detection of deformable obstacles such as tall grass [11]. Other researchers have used laser range sensors mounted on a ground vehicle to estimate a traversability cost for paths through natural terrestrial terrain, then fit a regression model based on color overhead imagery to predict the traversability costs for terrain over the horizon [12]. However no research has addressed the detection of non-geometric obstacles where their appearance has not been known *a priori*.

This paper addresses the issue of generalizing local terrain sensing by training a visual classifier to recognize classes corresponding to different outputs of the local sensors. In the context of mobility, local sensors identify a terrain class, and a visual classifier identifies where that terrain appears in the distance. For this work, local terrain was classified based on vibrations in the rover structure, and distant terrain was classified as belonging to one of the locally identified classes. Experiments were performed using a four-wheeled rover in Mars-analog rough terrain. Results demonstrate the accuracy of this classifier at identifying the locally sensed terrain class

This work was supported by the NASA Jet Propulsion Laboratory (JPL) through the Mars Technology Program.

in the distance. This paper represents an update of the work described in [13], presenting results from more recent experiments with longer data sets and improved range data outlier rejection.

Section 2 describes the self-supervised classification approach and the architecture of the component classifiers. Section 3 gives details about the experiments. Section 4 presents the results of the experiments. Section 5 describes the conclusions drawn from these results and suggests directions for future research.

II. Approach

A. Overview

In this paper, self-supervised classification is accomplished using a previously trained (i.e. supervised) vibration-based terrain classifier which provides labels for on-line training of a visual classifier, as illustrated in Fig. 1(a). Once labeled training data is available, standard supervised classification techniques can be used to train the visual classifier. The visual classifier is then able to predict the class of the terrain in the distance, as illustrated in Fig. 1(b). Significant challenges of such an approach lie in ensuring that the automated training process relies on correctly labeled training images, and that it does not overtrain the visual classifier. Details of the component vibration-based and vision-based classifiers are presented in this section, followed by details of the self-supervised training approach.

B. Local Vibration-Based Terrain Classification

Here a supervisory classifier (which performs the initial labeling of terrain patches) relies on features drawn from the vibration signature arising from rover wheel-terrain interaction. The vibrations are recorded by a contact microphone fixed to the rover suspension near its right-front wheel. This sensory mode was proposed in [14], and classification results were presented in [15]. Similar work has been presented for high-speed ground vehicles in [16]. This approach is attractive because it relies on features arising from physical rover-terrain interaction, and thus provides useful cues for rover mobility prediction.

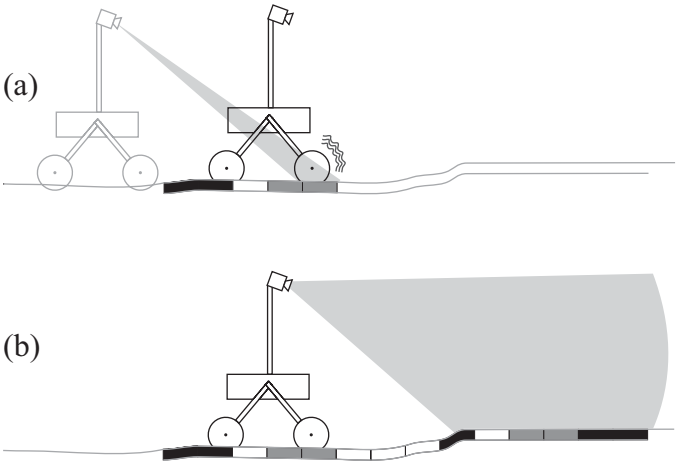

Fig. 1. Schematic of self-supervised classification. (a) Vibration-supervised training of visual classifier. (b) Prediction using visual classifier.

The data features used by the vibration-based classifier are identical to those used in [15]. Vibrations are represented as the log-scaled magnitudes of the power spectral density of the vibration time signal. The power spectra represent 557 frequencies ranging from 20.5 Hz to 12 kHz, estimated using Welch's method [17]. Here the magnitudes are averaged over 1-second windows to reduce noise. While 12 kHz is very high compared to the frequencies normally associated with vehicle vibrations, such high frequencies are observed when sensing rigid wheels surmounting rocks.

The 557-element vector representing the vibration data is used as an input to a support vector machine (SVM) classifier [18]. Support vector machines are well established classifiers which use kernel functions to measure the similarity between new data and training data. For vibration-based classification, a second-order polynomial kernel is used, because it was empirically observed to demonstrate the highest classification accuracy in previous experiments. For multi-class classification, the probability estimation approach described in [19] is used. This yields a class likelihood for each terrain class. For this work, SVM algorithms are implemented in C++ using the LIBSVM library, with custom optimizations for classification using a linear kernel [20].

C. Remote Vision-Based Classification

The local vibration-based classifier labels terrain patches as belonging to one of several *a priori* defined classes, and the images associated with these patches are employed to train a vision-based terrain classifier in a self-supervised framework. The vision-based classifier labels terrain based on images collected using a forward-looking color stereo camera pair. Vision-based classification has been studied extensively for terrain classification (e.g. [21,22]). Features observable using stereo cameras include color, visual texture, and geometry. These features are represented as inputs to the vision-based classifier as described below.

Color data is directly available from the cameras as red, green, and blue (RGB) intensities. However, overall illumination intensity affects all three values in a raw RGB representation and is known to yield poor classification results, so the hue, saturation, and value (HSV) representation is used as in [12]. Here hue (an angle) is represented as two values, sin(hue) and cos(hue), to eliminate the artificial discontinuity at 2π. Thus, color is represented as a 4-element vector: [sin(hue), cos(hue), saturation, value].

Visual texture is a measure of the local spatial variation in intensity in the image. Researchers have proposed many metrics for visual texture, including Gabor filter representations and local energy methods [23,24]. This work uses a wavelet-based approach, similar to the one demonstrated in [25]. Here the grayscale image is decomposed

with the Haar wavelet. Three scales of wavelets are used, each scale having horizontal, diagonal, and vertical (HDV) wavelets, corresponding to estimating the derivative in the horizontal, diagonal, and vertical directions at each length scale. The scales used are 2, 4, and 8 pixels. Because this data is noisy, the magnitudes of the wavelet coefficients are averaged over windows of 7, 9, and 11 wavelets. Thus, visual texture is represented by a 9-element vector, composed of the window-averaged horizontal, diagonal, and vertical wavelet coefficients at each scale.

Geometric terrain data is available for the terrain through stereo image processing. This raw output is a point cloud. For this work, we consider the terrain to be a grid of 20-cm by 20-cm patches, and calculate the statistics of the points within each grid. The first element of the geometric feature vector is the average slope of the terrain, defined as the angle the least-squares-fit plane makes with the horizontal. The second element is the mean-squared deviation from that plane along its normal. The third element is the variance in height of the pixels, and the fourth is the height difference between the highest and lowest pixels in the patch. Thus, the geometry of each patch is represented as a 4-element vector: [slope, plane fit deviation, height variance, height range].

It should be noted that because the range data output from the stereo processing algorithm is noisy, a simple approach is used to reject obvious outliers prior to extracting the geometric features. A loose 2½-D restriction is applied as follows. Iterating from the bottom to the top of the stereo image, a point (i.e. a pixel with range data) is rejected if it is 0.5 meters closer to the camera than the closest point in the five rows below it. Then, iterating from the top to the bottom of the image, a point is rejected if it is 0.5 meters farther from the camera than the farthest point in the five rows above it. This process has been observed to eliminate most of the obviously incorrect range data.

As with the vibration-based classification, the vision-based classification employs a support vector machine classifier. A straightforward approach of concatenating the color, visual texture, and geometric features into a single feature vector resulted in poor classifier accuracy, so naïve Bayes fusion is used for the results presented here. This approach assumes that color, visual texture, and geometric features are conditionally independent given the terrain class. Thus, the likelihood of a terrain patch belonging to a terrain class is the product of the class likelihoods for each sensing mode. Note that since there may be many pixels observed in each terrain patch, the overall estimate of the class likelihood for the color mode is taken as the geometric mean of the individual pixel class likelihoods. The same approach is used for visual texture.

To estimate the class likelihoods for each sensing mode, a support vector machine is used, as with the vibration data. Training is done separately for each sensing mode. For this SVM, linear and low-order polynomial kernels are considered, because they can be computed efficiently independent of the amount of training data. This is especially important for the color and texture modes because the classifier is applied to each pixel individually, so classification of each pixel must be very rapid.

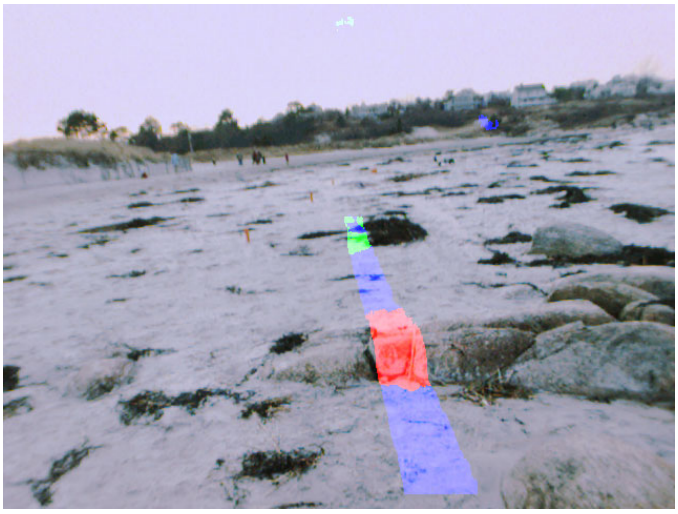

Fig. 2. Sample image from stereo camera with future path of rover marked in blue (sand), red (rock), and green (beach grass)

D. Self-Supervised Training Framework

Here, the visual data needed for training the vision-based classifier is collected using forward-looking stereo cameras. A sample image from one of the stereo cameras is shown in Fig. 2. Because the cameras cannot see the terrain immediately beneath the rover's wheels, stereo images of terrain patches are stored in memory and recalled when the rover traverses them. This relies heavily on stereo processing to identify the distance to patches in the image and on accurate position estimation (over 1-10m distances) to identify when the rover has reached a patch of terrain. It has the advantage that each terrain patch is typically observed from multiple distances, which helps to make the classification independent of range.

Training data is accumulated as the rover travels, with a separate set of training data stored for each terrain. To ensure that training time for the visual classifier is less than the stereo frame rate (not including feature extraction), each terrain is limited to a maximum of 400 data points—"old" data is forgotten if new data arrives that would exceed that maximum. Classification of the terrain is implemented on a patch level, based on naïve Bayes fusion of the class likelihoods of the color, visual texture, and geometric sensing modes.

III. Experiment Details

This self-supervised classification approach is compared to traditional manually trained (supervised) classification using experimental data collected on TORTOISE, a four-wheeled mobile robot developed at MIT, in Mars-analog outdoor terrain.

Fig. 3. TORTOISE rover, showing section enlarged in Fig. 4

A. Robot Configuration

TORTOISE, shown in Figs. 3 and 4, is an 80-cm-long, 50-cm-wide, 90-cm-tall robot with 20-cm-diameter rigid aluminum wheels with grousers. The wheels on either side are connected to the main body and mast via a differential. The robot is outfitted with a forward-looking mast-mounted stereo camera pair, a belly-mounted monocular camera, a vibration sensor, and a body-mounted two-axis tilt sensor. Forward-looking images were captured with the stereo camera pair, a Videre Design "dual DCAM" capable of capturing color images with 640 x 480 resolution. Range data were extracted from the stereo images using Videre Design's commercial stereo processing software [26]. Color images of the right-front wheel and its immediate surroundings were captured using the belly-mounted monocular camera. Vibrations from the right-front wheel were sensed using a Signal Flex SF-20 contact microphone. Body pitch and roll were measured with a Crossbow CXTA02 two-axis tilt sensor. Additional sensors include a torque sensor and suspension configuration sensors, though these sensing modes are not used in this work.

During experiments, the rover traveled at an average speed of 3 cm/sec. Vibrations were recorded at 44.1 kHz, body pitch and roll were captured at 25 Hz, images from the belly-mounted camera were captured at 2 Hz, and forward-looking stereo images were captured every 2 seconds. These data were stored during experiments and processed offline.

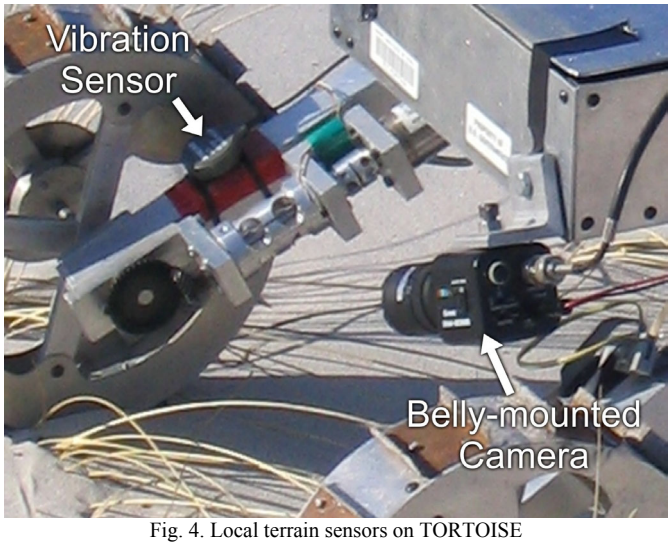

Fig. 4. Local terrain sensors on TORTOISE

B. Experiment Environment

Experiments were performed at Wingaersheek Beach in Gloucester, MA. This is a sandy beach with a mixture of small and large rock outcrops, relative to the size of the rover, and loose rocks. This site was chosen due to its similarity in appearance to the MER landing sites on Mars. In this environment, sand and rock were considered to be distinct terrain classes. To demonstrate the ability of the classification approach in a multi-class setting, matted piles of beach grass were used as a third terrain class. A sample scene is shown in Fig. 5. In this image, sand appears as a uniform gray flat surface, rock appears tan and orange with some steep slopes and fine uniform texture, and beach grass appears highly textured with mixed browns and dark shadows.

Experiments used in this analysis were conducted over three days. Each data set was collected during a single rover

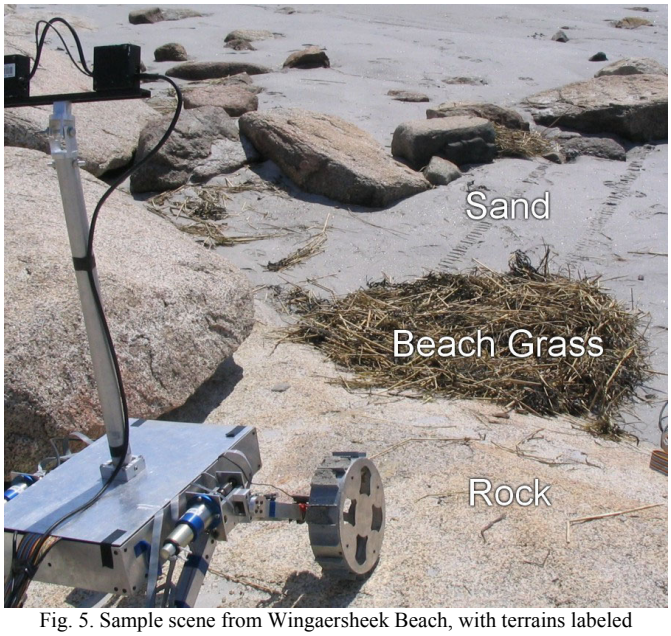

Fig. 5. Sample scene from Wingaersheek Beach, with terrains labeled

traverse 10 to 35 meters along a straight-line path containing a combination of two or three terrains. No two paths were identical. During the experiments cloud cover ranged from overcast to direct sun.

C. Data Processing

Data were collected during the experiments and stored for processing offline. In each data set images were hand-labeled to identify a ground truth terrain class for each pixel. Each image typically contained regions corresponding to two or three of the terrain classes. For each data set, between 8 and 25 evenly-spaced images were hand labeled for testing. Separate hand-labeled images from the beginning of each data set were used for training the manually trained vision-based classifiers. In all, six data sets, with 93 hand-labeled images, were used to assess the accuracy of the visual classifiers.

For self-supervised classification, the vibration-based classifier was trained using hand-labeled vibration data from the data sets not being tested. The true terrain was identified manually based on images from the belly-mounted camera.

IV. Results

The accuracy of the self-supervised classifier was assessed for each of the data sets as described in section III. Fig. 6 shows the receiver operating characteristic (ROC) curves for a representative data set. Here, the horizontal axis indicates the false positive percentage (%FP) (for example, instances when a patch of sand or beach grass was falsely identified as rock) and the vertical axis indicates the true positive percentage (%TP). Each terrain forms a curve on the plot as the threshold for leaving terrain "unclassified" is adjusted. Note that the scale of the horizontal axis is magnified 10x relative to the vertical axis. In this plot it can be seen that both sand and beach grass are classified with very high accuracy: 80% of the patches for each of these terrains is positively identified before 1% of the remaining terrains are misidentified. The accuracy of rock classification is relatively poor—20% of the rock was identified with 2% false positive detection—however this was due to a relatively low incidence of rock in this particular data set, which yielded a paucity of training data. Beyond the identified 20% of rock patches, the classifier predicted that the terrain was more likely to belong to one of the other classes.

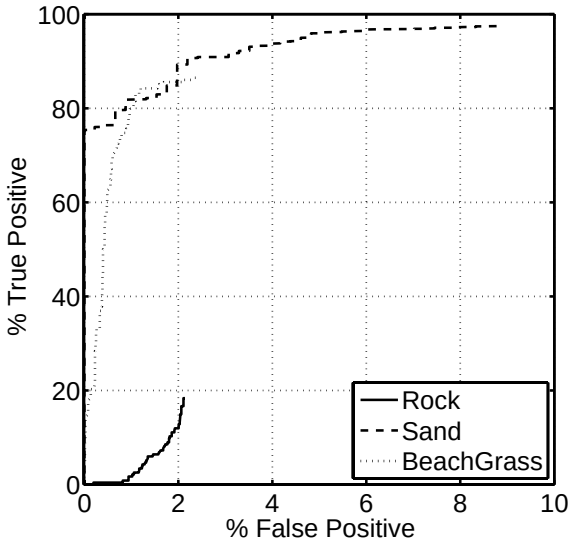

Fig. 6. ROC curves for self-supervised visual classifier

For comparison, a manually trained classifier was implemented. This classifier was trained using color, texture, and geometry data from hand-labeled images taken from the start of the same rover traverse as the test images. For each terrain 400 data points for each sensing mode were randomly selected and used to train a SVM classifier. Note that it is expected that the manually trained classifier would perform at least as well as the self-supervised classifier. ROC curves for this classifier applied to the same test images as used for the self supervised classifier are shown in Fig. 7. Here it may be seen that the manually trained classifier performed very well at identifying both sand and beach grass. Nearly 95% of the sand patches were identified before any other terrain was falsely identified as sand. For beach grass, nearly 70% was identified with less than 1% of the remaining terrain falsely identified. Results for rock were also good, with 80% of the

TABLE I
COMPARISON OF SELF-SUPERVISED CLASSIFIER TO MANUALLY TRAINED CLASSIFIER

	Self-supervised Classifier	Manually trained Classifier	Manually trained Classifier (Prior Data Set)
Mean % True Pos. [95% Conf. Int.]	65.0% [43.5% - 86.5%]	76.6% [60.7% - 92.5%]	55.2% [35.3% - 75.1%]
St. Dev. of % True Pos.	43.5%	32.1%	40.1%
Mean %TP/(%TP + %FP) [95% Conf. Int.]	0.87 [0.78 – 0.95]	0.91 [0.84 – 0.98]	0.68 [0.49 – 0.86]
St. Dev. of %TP/(%TP + %FP)	0.14	0.13	0.36

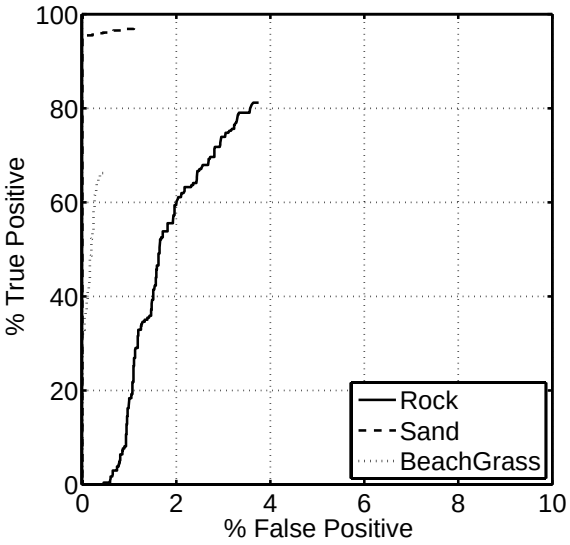

Fig. 7. ROC curves for manually trained visual classifier

rock positively identified before 4% of the remaining terrain was incorrectly identified as rock.

Self-supervised classifiers and manually trained classifiers were implemented for each of the six data sets and the results are shown in the first two columns of Table I. The top two rows show statistics of the true positive percentage of the classifiers when no data is left unlabeled, corresponding to the vertical coordinate of the ROC curve endpoints. The bottom two rows show statistics related to the ratio between the true positive percentage and the false positive percentage. The metric, %TP/(%TP + %FP), is closely related to the fraction of labeled patches which are labeled correctly.

In this table it can be seen that the manually trained classifier slightly outperforms the self-supervised classifier in terms of the mean true positive percentage, however the difference is still within the bounds of the 95% confidence interval. In addition, there is significantly more variability in the true positive percentile with the self-supervised classifier. The more surprising result is that in terms of the mean %TP/(%TP + %FP), the self-supervised classifier performs nearly as well as the manually trained classifier, which should be seen as the upper limit of a visual classifier of this type (i.e. using a support vector machine with this visual feature representation).

It should be noted, however, that the self-supervised approach is intended for situations when a manually trained classifier is not a viable option. Thus, the accuracy of a self-supervised classifier is more fairly compared to a manually trained classifier trained on the previous data set, because the time it takes to manually label data would delay the classifier's implementation. The accuracy of such a classifier is shown in the third column of Table 1. Under these conditions it can be seen that the self-supervised classifier outperforms the manually trained classifier across all metrics. In particular, it was observed that the manually trained classifier performance was particularly degraded for the first data set of each day, when the classifier was trained using visual data from the previous day. This suggests that the particular lighting conditions are critical for accurate classification, so that even the imperfect and limited supervision of the vibration-based classifier is preferable to "perfect" training under different conditions.

A. Impact of Supervisory Classifier Accuracy

An additional factor that can affect the performance of the self-supervised classification approach is the accuracy of the supervising classifier, in this case the vibration-based terrain classifier. In scenarios where a limited amount of training data is present, there is a tradeoff between the number of terrain patches used for training and the confidence the vibration has in the labels for those patches. To study this effect, the classification confidence threshold was varied above and below the nominal value (0.95) and ROC curves were generated for each threshold.

Preliminary results demonstrate a complex interaction between the vibration-based classifier threshold and the accuracy of the resulting vision-based classifier. Sample results are shown in Figs. 8 through 10. Fig. 8 shows the effect of the varied vibration classifier threshold on the classification of sand in the third data set. Here the classification accuracy (as observed in the proximity to the upper left corner of the plot) is highest for the low confidence threshold (0.75), decreases as the threshold is raised (0.90), and then improves again with a sufficiently high confidence threshold (0.975 and 0.995). This suggests that most of the classifications at lowest threshold were correct, and that valid training data was being discarded as the threshold was increased. The improvement at the high-threshold values suggests that some incorrect training data was discarded.

Fig. 9 demonstrates a different behavior. Here it is seen that

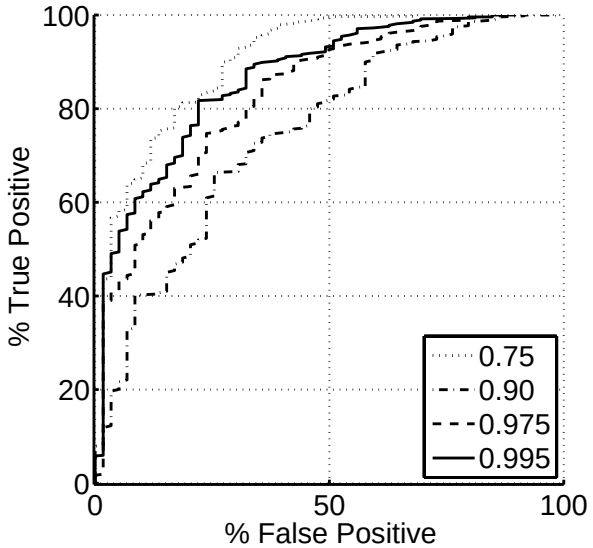

Fig. 8. Sand ROC curve as function of vibration-based classifier threshold for data set 3

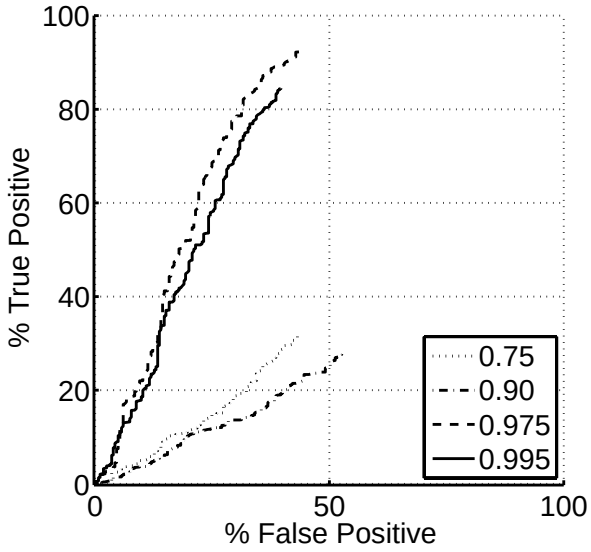

Fig. 9. Sand ROC curve as function of vibration-based classifier threshold for data set 4

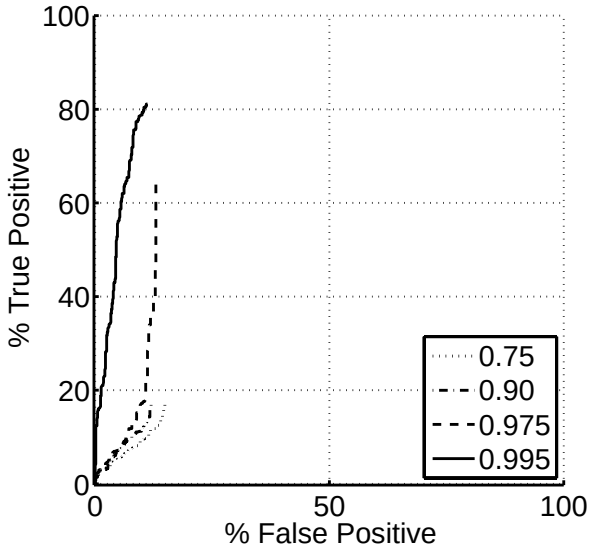

Fig. 10. Sand ROC curve as function of vibration-based classifier threshold for data set 6

the classifiers trained using the low confidence threshold (0.75 and 0.90) were significantly less accurate than the high-threshold classifiers. Here it may be assumed that a significant amount of incorrect training data was degrading the behavior of the classifiers, and that once that data was discarded, the classification accuracy improved dramatically.

Fig. 10 demonstrates a third behavior: the gradual improvement of the classification accuracy as a function of the classification threshold. Here, the low-threshold classifier recognizes a small fraction of the sand. As the threshold is increased, the classifier is able to recognize a larger fraction of the sand without increasing the rate of false positives.

These plots demonstrate the importance of high classifier confidence in the supervising classifier. Though the behavior illustrated in Fig. 8 shows that occasionally the low-threshold classifier outperforms the high-threshold classifier, the strong improvements demonstrated in the other figures suggests that the high-threshold classifier may be more likely to give good classification results, as would be expected.

B. Computation time

An effort was made to limit the computational complexity of training and testing in the proposed framework. As a result, the most computationally intensive tasks are the stereo data extraction and the texture feature computation. Extraction of the geometric features from the 3-D point cloud takes an average of 5 seconds per image using a Matlab script on a Pentium 1.8 GHz desktop PC. Texture feature extraction takes 17.3 seconds using an unoptimized Matlab script. A C-code implementation would be expected to run much faster.

Training the self supervised classifier took 1.5 seconds on average on a Pentium 1.8 GHz desktop PC, using the LIBSVM library. Classification was slightly slower, 4.1 seconds per image on average, once the geometric and texture features were extracted. Since the classification results from the sensing modes were combined using a Matlab script, it would be expected that a C-code implementation of the full classification would run slightly faster.

V. Conclusions

In this work, a self-supervised classification approach was presented which automatically trains a visual classifier based on terrain classes sensed locally using a vibration-based terrain classifier. It employs a memory of the appearance of terrain currently being traversed to supply training data for the visual classifier. This approach was shown to perform nearly as well as a manually trained classifier on an outdoor Mars-analog data set. Since the manually trained classifier was provided with correct class labels for large regions of images from the same traverses, it represents the limiting accuracy which the self-supervised classifier could be expected to achieve.

It was also demonstrated that the self-supervised classifier outperformed a manually trained classifier trained using hand-labeled images from the previous data set. This suggests that self-supervised classification may be preferred over a manually trained classifier when the illumination is changing more quickly than a manually trained classifier can be re-trained.

Additionally, the tradeoff between the number of training patches and the confidence in their labels was also studied. It was observed that, in general, maintaining a high confidence threshold on the training labels tended to yield more accurate vision-based classification results.

Finally, it should be noted that even with relatively long traverses, some terrains had few terrain patches sensed using the vibration-based classifier, resulting in poor visual classifier training. An increase in the classifier accuracy would likely be gained by increasing the length of the traverses

further. Methods for automatically analyzing the adequacy of visual classifier training is a focus of current work.

A. Future Work

Since the overarching goal of this research is to be able to predict the traversability properties of the terrain, a logical next step would be to incorporate measurements of the terrain properties into the terrain class definitions. Additionally, for situations when the vibration signatures of various terrain classes are not known *a priori*, autonomous segmentation of terrain into distinct terrain classes based on visual and tactile cues is necessary.

REFERENCES

[1] Nilsson, N. J. *Principles of Artificial Intelligence*, Tioga Publishing Company, Palo Alto, CA. 1980.

[2] Stentz, A. "The D* algorithm for real-time planning of optimal traverses" Technical Report CMU-RI-TR-94-37, Robotics Institute, Carnegie Mellon University, October 1994.

[3] Goldberg, S., Maimone, M., and Matthies, L. "Stereo vision and rover navigation software for planetary exploration" in *Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana, March 2002.

[4] Talukder, A., Manduchi, R., Castaño, R., Owens, K., Matthies, L., Castano, A., Hogg, R. "Autonomous Terrain Characterization and Modeling for Dynamic Control of Unmanned Vehicles" in *Proceedings of the IEEE Conference on Intelligent Robots and Systems* (IROS), Lausanne, Switzerland, September 2002.

[5] Wilcox, B. "Non-geometric hazard detection for a Mars Microrover", in *Proceedings of the AIAA Conference on Intelligent Robotics in Field, Factory, and Space*, Vol. 2, 1994.

[6] Brooks, C., and Iagnemma, K. "Visual wheel sinkage measurement for planetary rover mobility characterization," *Autonomous Robots*, Vol. 21, pp. 55-64. DOI 10.1007/s10514-006-7230-9.

[7] Iagnemma, K., Shibly, H., and Dubowsky, S. "On-line terrain parameter estimation for planetary rovers." in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation* (ICRA 02), Washington, DC, May 2002.

[8] Reina, G., Ojeda, L., Millella, A., and Borenstein, J. "Wheel slippage and sinkage detection for planetary rovers," *IEEE Transactions on Mechatronics, Special Issue on Novel Aspects in Robotics*, Vol. 11, No. 2, April 2006, pp. 185-195.

[9] Kang, S. "Terrain parameter estimation and traversability assessment for mobile robots" MS Thesis, Department of Mechanical Engineering, MIT, 2003.

[10] Angelova, A., Matthies, L., Helmick, D., and Perona, P. "Learning slip behavior using automatic mechanical supervision," in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation* (ICRA), Rome, Italy, April 10-14, 2007.

[11] Kim, D., Sun, J., Oh, S. M., Rehg, J. M., and Bobick, A. F. "Traversability Classification using Unsupervised On-line Visual Learning for Outdoor Robot Navigation" in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, Florida. May, 2006.

[12] Sofman, B., Lin, E., Bagnell, J. A., Vandapel, N., Stentz, A. "Improving Robot Navigation Through Self-Supervised Online Learning" in *Proceedings of Robotics: Science and Systems*, August 2006.

[13] Brooks, C. A., and Iagnemma, K. D. "Self-Supervised Classification for Planetary Rover Terrain Sensing," in *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, March 3-10, 2007, Paper 1164.

[14] Iagnemma, K. D. and Dubowsky, S. "Terrain estimation for high speed rough terrain autonomous vehicle navigation," in *Proceedings of the SPIE Conference on Unmanned Ground Vehicle Technology IV*, Orlando, FL, 2002, Paper 4715-31.

[15] Brooks, C., and Iagnemma, K. "Vibration-based terrain classification for planetary rovers." *IEEE Transactions on Robotics*, Vol. 21, No. 6., pp. 1185-1191, December 2005.

[16] Sadhukhan, D., Moore, C., and Collins, E. "Terrain estimation using internal sensors," in *Proc. of the IASTED Int. Conf. Robot. Appl.*, Honolulu, HI, 2004, Paper 447-800.

[17] Welch, P. D. "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. AU-15, pp. 70-73, June 1967.

[18] Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer: New York. 1995.

[19] Wu, T.-F., Lin, C.-J., and Weng, R. C. (2004). "Probability estimates for multi-class classification by pairwise coupling" *Journal of Machine Learning Research*, 5:975–1005.

[20] Chih-Chung C., and Chih-Jen L. (2001). LIBSVM: a library for support vector machines. Software retrieved January, 2006 available at *http://www.csie.ntu.edu.tw/~cjlin/libsvm*.

[21] Rasmussen, C. "Combining Laser Range, Color, and Texture Cues for Autonomous Road Following", in *Proceedings of the 2002 IEEE International Conference on Robotics & Automation*, Washington, DC, May 2002.

[22] Shi, X., and Manduchi, R. "A Study on Bayes Feature Fusion for Image Classification," IEEE Workshop on Statistical Algorithms for Computer Vision.

[23] Bouman, C. and Liu, B. "Multiple Resolution Segmentation of Textured Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 2, pp. 99-113.

[24] Reed, T., and Hans du Buf, J. "A review of recent texture segmentation and feature extraction techniques," *CVGIP: Image Understanding*, Vol. 57(3), May 1993, pp. 359-372.

[25] Espinal, F., Huntsberger, T., Jawerth, B., Kubota, T. "Wavelet-Based Fractal Signature Analysis for Automatic Target Recognition," *Optical Engineering*, Vol. 37, No. 1, pp166-174, January 1998.

[26] Videre Design. Retrieved on May 25, 2006 at *http://www.videredesign.com/index.htm*.

Christopher Brooks is a graduate student in the Mechanical Engineering department of the Massachusetts Institute of Technology. He received his B.S. degree with honor in engineering and applied science from the California Institute of Technology in 2000, and his M.S. degree from the Massachusetts Institute of Technology in 2004. His research interests include mobile robot control, terrain sensing, and their application to improving autonomous robot mobility. He is a member of Tau Beta Pi.

Karl Iagnemma is a principal research scientist in the Mechanical Engineering department of the Massachusetts Institute of Technology. He received his B.S. degree *summa cum laude* in mechanical engineering from the University of Michigan in 1994, and his M.S. and Ph.D. from the Massachusetts Institute of Technology, where he was a National Science Foundation graduate fellow, in 1997 and 2001, respectively. He has been a visiting researcher at the Jet Propulsion Laboratory. His research interests include rough-terrain mobile robot control and motion planning, robot-terrain interaction, and robotic mobility analysis. He is author of the monograph *Mobile Robots in Rough Terrain: Estimation, Motion Planning, and Control with Application to Planetary Rovers* (Springer, 2004). He is a member of IEEE and Sigma Xi.